

# T.I.P MART

T.I.P MART is a modern Android marketplace application designed specifically for students of the Technological Institute of the Philippines (T.I.P). Built with Kotlin and Jetpack Compose, it provides a seamless platform for students to buy and sell items within their campus community.

## Features

### Authentication and User Management

- Secure Login and Signup with Firebase Authentication
- Profile Management with verification status
- Multi-campus Support (Manila, Quezon City)

### Shopping Experience

- Product Browsing with search and filtering
- Shopping Cart with quantity management
- Wishlist functionality for favorite items
- Secure Checkout with multiple payment methods
- Order Tracking with real-time status updates

### Seller Features

- Product Listing with image upload
- Inventory Management
- Order Management with pickup scheduling
- Sales Dashboard and analytics
- Customer Communication through in-app messaging

### Smart Pickup System

- Campus-based Pickup locations
- Flexible Time Scheduling (8 AM - 8 PM)
- Pickup Code Generation for secure handovers
- Real-time Notifications for order updates

### Payment and Security

- Multiple Payment Methods (GCash, Maya, Bank Transfer, Cash)
- Secure Transactions with Firebase integration
- Order Dispute Resolution system
- Proof of Pickup with image verification

### Communication

- In-app Messaging between buyers and sellers

- Push Notifications for important updates
- Rating and Review System for trust building

## Tech Stack

### Frontend

- Kotlin - Primary programming language
- Jetpack Compose - Modern UI toolkit
- Material Design 3 - UI components and theming
- Navigation Component - App navigation
- Hilt or Dagger - Dependency injection

### Backend and Services

- Firebase Authentication - User management
- Cloud Firestore - Real-time database
- Firebase Storage - Image and file storage
- Firebase Cloud Messaging - Push notifications

### Architecture

- MVVM Pattern - Clean architecture
- Repository Pattern - Data layer abstraction
- Coroutines - Asynchronous programming
- StateFlow or LiveData - Reactive programming

## Getting Started

### Prerequisites

- Android Studio Arctic Fox or later
- Android SDK 21+
- Firebase project setup
- Git

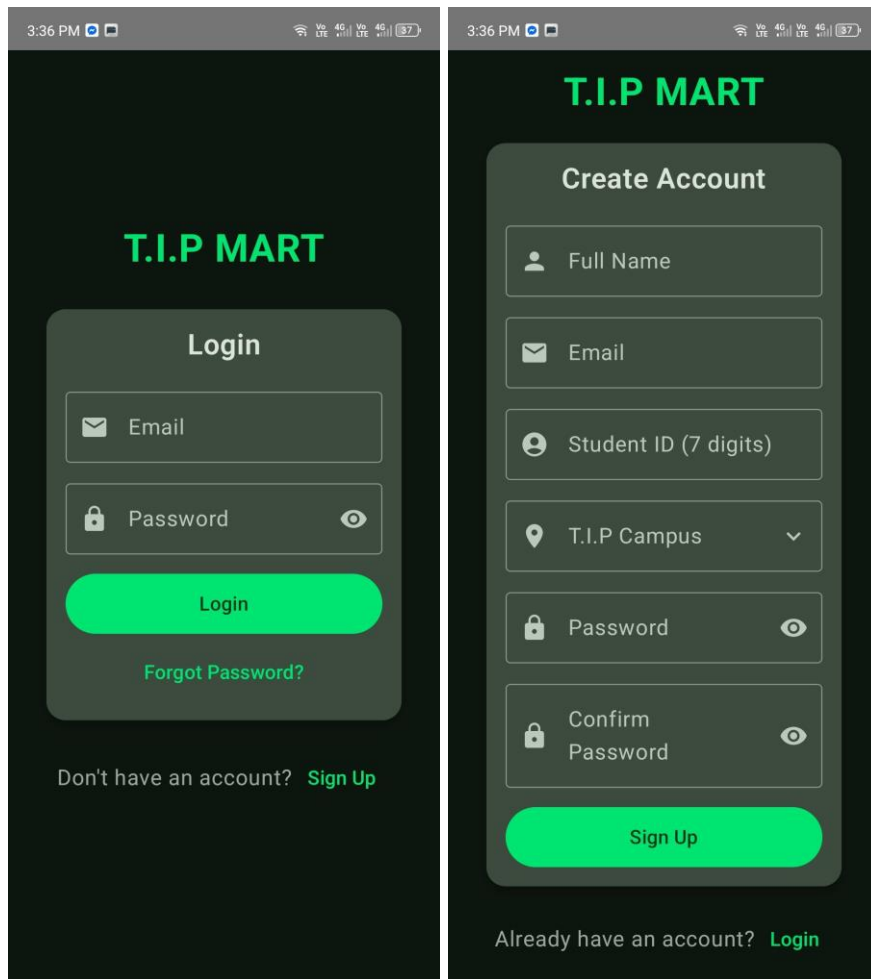
### Installation

1. Clone the repository  
git clone <https://github.com/yourusername/tip-mart.git>  
cd tip-mart
2. Open in Android Studio
  - Launch Android Studio
  - Select "Open an existing project"
  - Navigate to the cloned directory
3. Firebase Setup
  - Create a new Firebase project at [firebase.google.com](https://firebase.google.com)
  - Add your Android app to the project

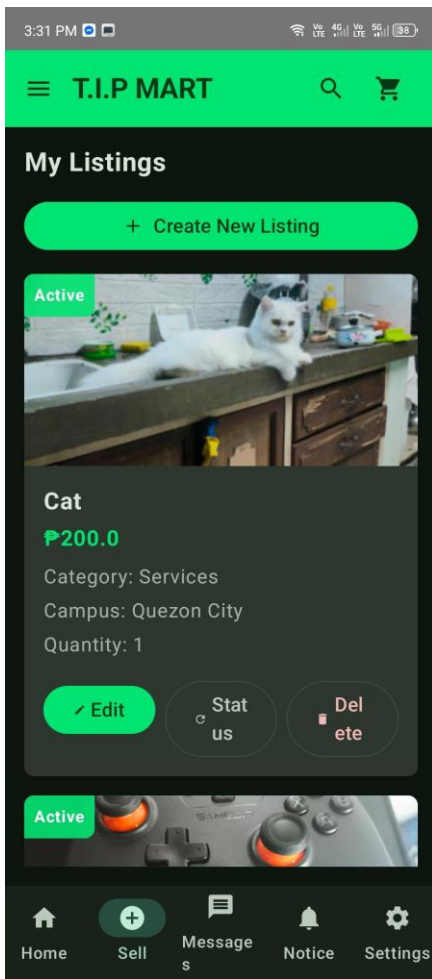
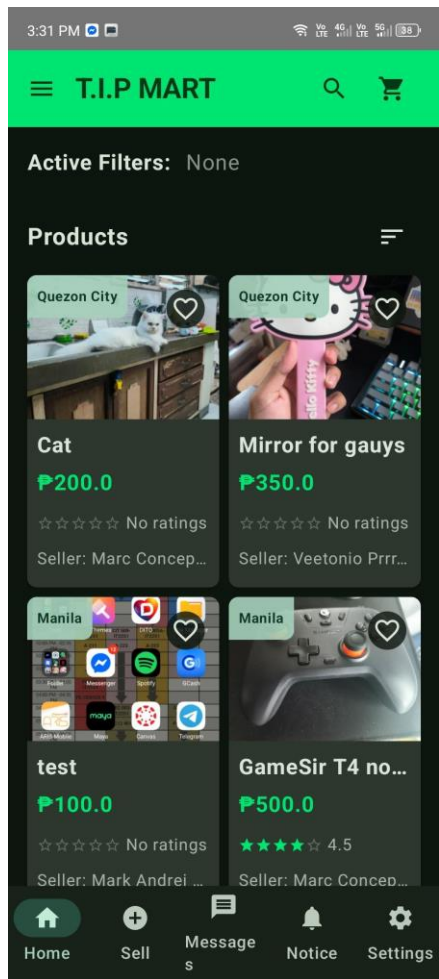
- Download google-services.json and place it in the app directory
  - Enable Authentication, Firestore, and Storage in Firebase Console
- 4. Build and Run
  - ./gradlew build
    - Connect your Android device or start an emulator
    - Click "Run" in Android Studio

## Key Screens

- Dashboard - Main hub with featured products
- Product Search - Browse and filter products
- Shopping Cart - Manage selected items
- Checkout - Secure payment processing
- Orders - Track order status and history
- Wishlist - Save favorite products
- Profile - User account management
- Messages - Communication center
- Seller Dashboard - Manage listings and sales



Login And Sign Up page



Home and Sell Page

3:32 PM

4G

5G

38

←

Create Listing

Product Images

Add Image

Product Title

Description

₱ Price (₱)

Quantity

1

Category

▼

Campus

▼

Create Listing

3:33 PM

4G


5G

37

←

Product Details

1



Mirror for gauys

₱350.0

☆☆☆☆☆ No ratings yet

V

Veetonio Prrince

Campus: Quezon City

Description

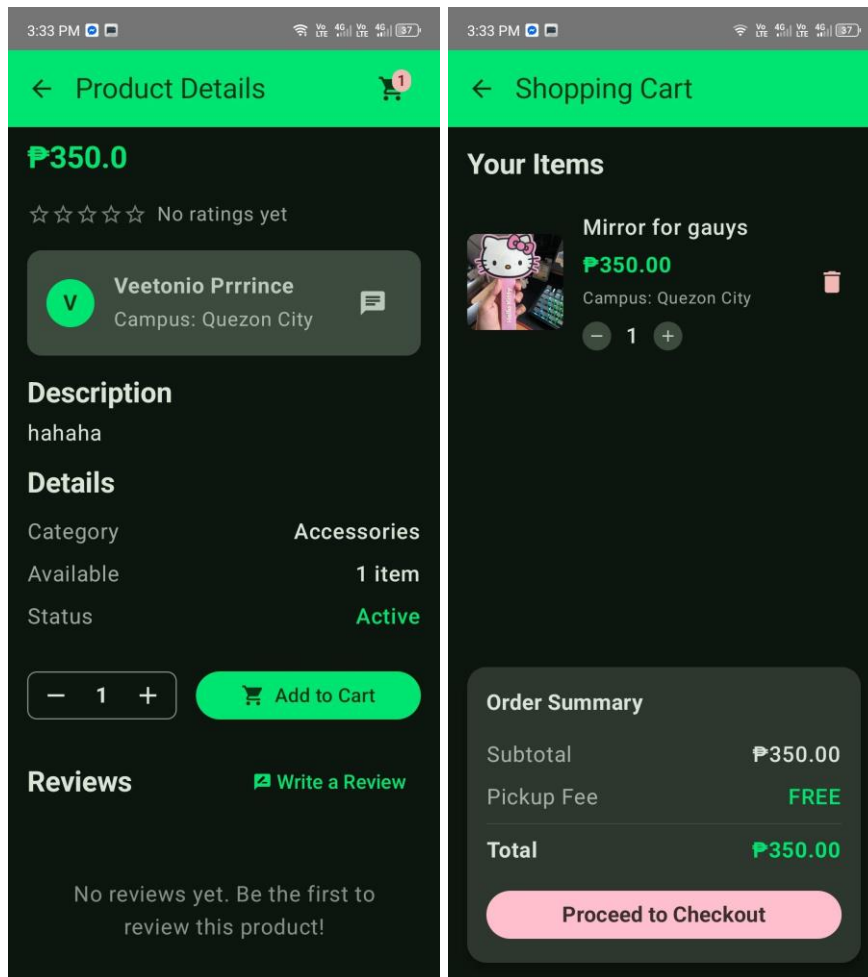
hahaha

Details

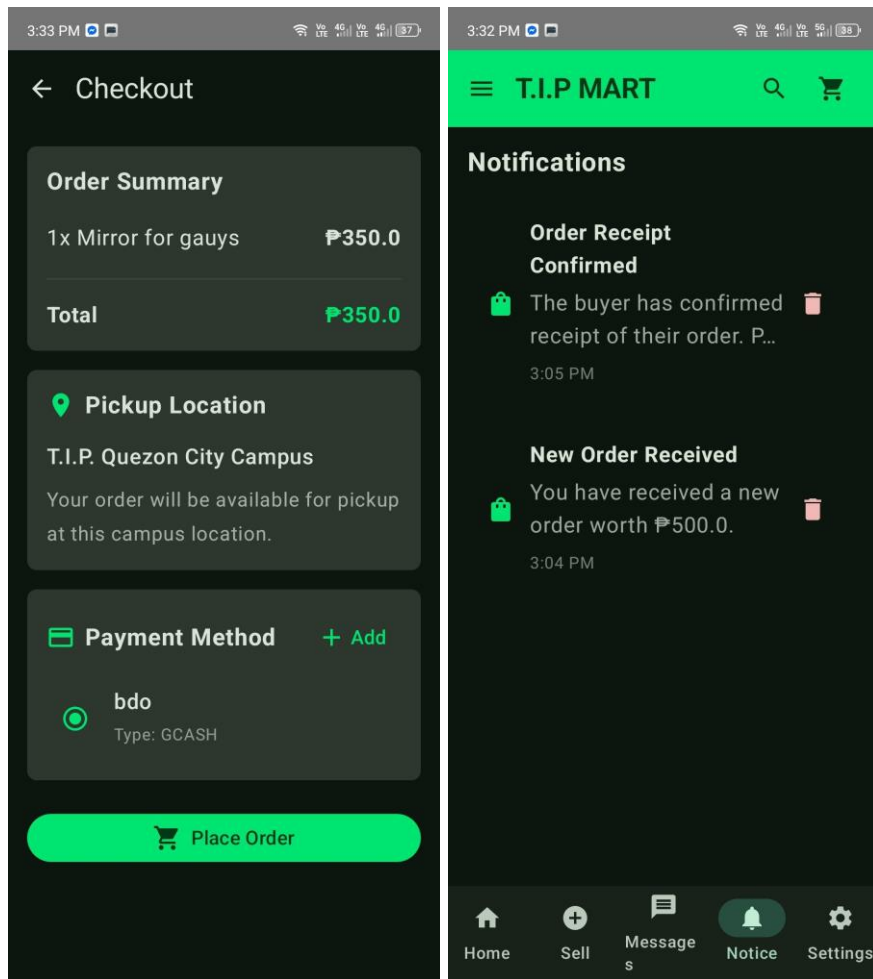
Category

Accessories

Create Listing and Product Details page



Product Listing and Shopping Cart Page

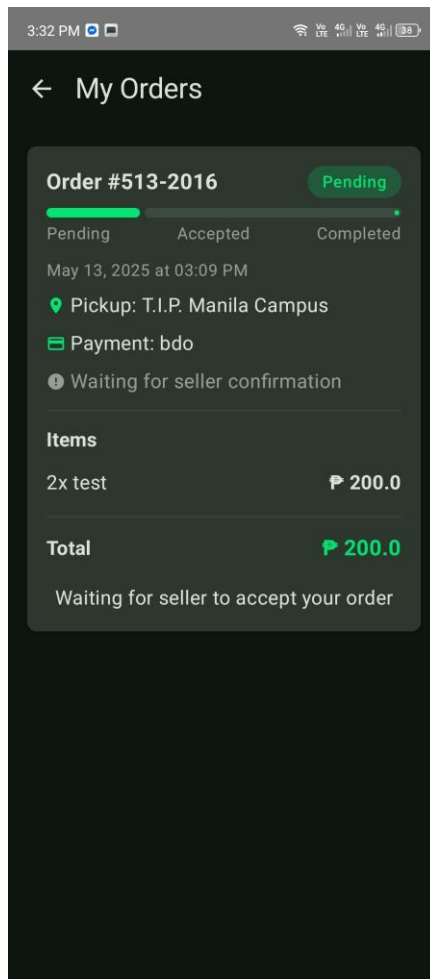
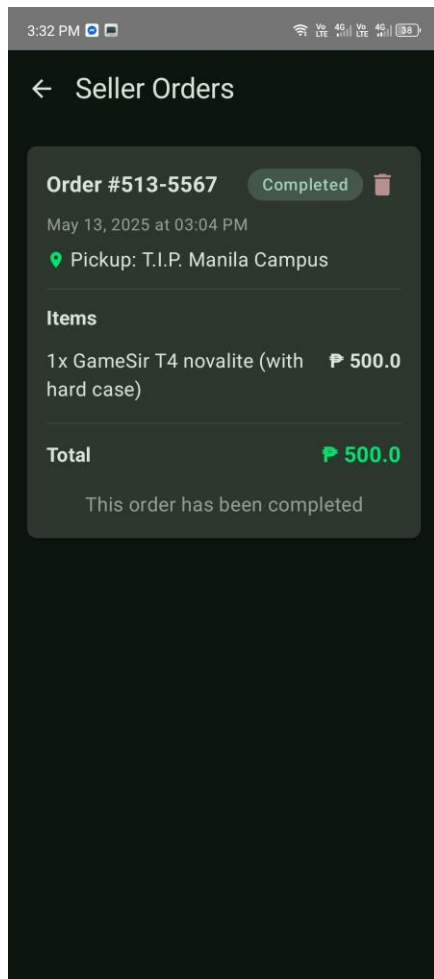


Check out and Notifications Page

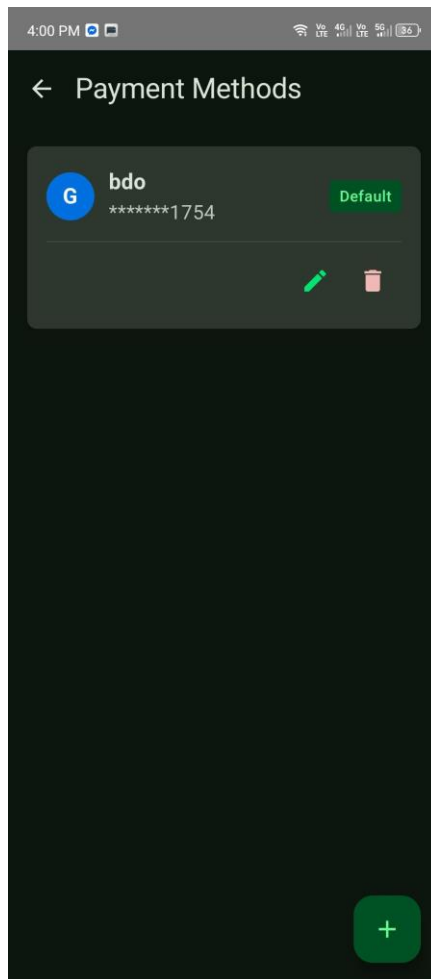
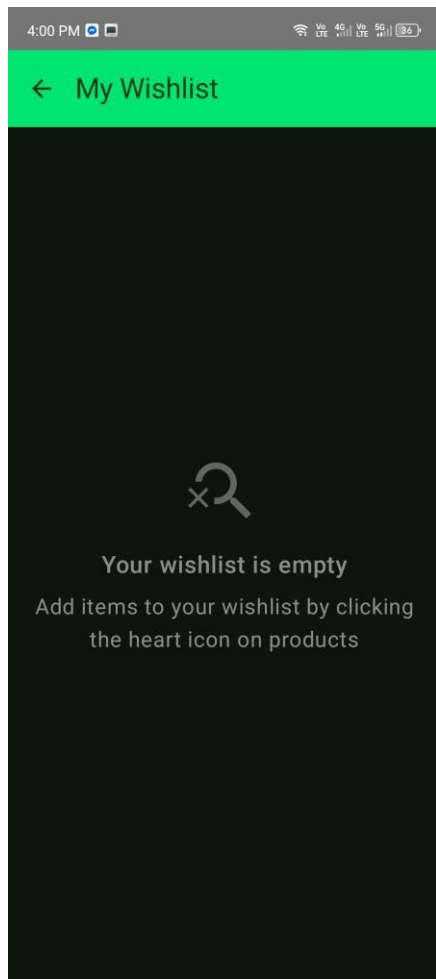




**Settings and Edit Profile Page**



**Seller and My Orders Page**



**WishList and Payment Methods page**